AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1-66 (Canceled).

67. (Currently amended) ~~An~~ A computer-implemented authenticated-encryption method that uses an n-bit block cipher, a key, and an n-bit nonce to encrypt a message into a ciphertext, the method comprising:

partitioning the message into m-1 message blocks and one final fragment, each message block having n bits and the final fragment having between 0 and n bits;

using the block cipher, the key, and the nonce to generate a sequence of m offsets, each offset having n bits, wherein the sequence of offsets is computed by (a) computing a 0^{th} basis offset by applying the block cipher, keyed by the key, to a constant; (b) for each positive number i, defining the i^{th} basis offset from the prior basis offset by shifting the prior basis offset left one position, and then xoring the resulting value with a constant that depends on the first bit of the prior basis offset; (d) computing a base offset by applying the block cipher, keyed by the key, to the xor of the 0^{th} basis offset and the nonce; (e) defining the 1^{st} offset in the sequence of offsets as the xor of the 0^{th} basis offset and the base offset; and (f) for each integer i between two and m, defining the i^{th} offset in the sequence of offsets as the xor of the prior offset and the j^{th} basis offset, where j is the number of zero-bits following the last one-bit when the number i is written in binary;

using the block cipher, the key, the nonce, and the length of the message to generate an n-bit final offset;

for each number i between 1 and m-1, xoring the i^{th} message block with the i^{th} offset to determine an i^{th} input block;

for each number i between 1 and m-1, applying the block cipher, keyed by the key, to the i^{th} input block, to determine an i^{th} output block;

for each number i between 1 and m-1, xoring the i^{th} output block with the i^{th} offset to determine an i^{th} ciphertext block;

concatenating the m-1 ciphertext blocks to determine a ciphertext body;

computing an encoded length by encoding the length of the final fragment as an n-bit string;

xoring the encoded length with the final offset to determine a precursor pad;

computing a pad by applying the block cipher, keyed by the key, to the precursor pad;

xoring the final fragment with a portion of the pad to determine a ciphertext fragment having the same length as the final fragment;

computing a padded ciphertext fragment by appending to the ciphertext fragment a sufficient number of zero bits so that the padded ciphertext fragment has n bits;

computing a checksum by xoring together the m-1 message blocks, the pad, and the padded ciphertext fragment;

computing a precursor full tag by xoring together the checksum and the m^{th} offset;

determining a full tag by applying the block cipher, keyed by the key, to the precursor full tag;

computing a tag as a portion of the full tag; and

defining the ciphertext to be the ciphertext body, the ciphertext fragment, and the tag.

68. (Previously presented) A computer-readable storage medium storing instructions that when executed by a computer cause the computer to perform an authenticated-encryption method that uses an n-bit block cipher, a key, and an n-bit nonce to encrypt a message into a ciphertext, the method comprising:

partitioning the message into m-1 message blocks and one final fragment, each message block having n bits and the final fragment having between 0 and n bits;

using the block cipher, the key, and the nonce to generate a sequence of m offsets, each offset having n bits, wherein the sequence of offsets is computed by (a) computing a 0^{th} basis offset by applying the block cipher, keyed by the key, to a constant; (b) for each positive number i, defining the i^{th} basis offset from the prior basis offset by shifting the prior basis offset left one position, and then xoring the resulting value with a constant that depends on the first bit of the prior basis offset; (d) computing a base offset by applying the block cipher, keyed by the key, to the xor of the 0^{th} basis offset and the nonce; (e) defining the 1^{st} offset in the sequence of offsets as the xor of the 0^{th} basis offset and the base offset; and (f) for each integer i between two and m, defining the i^{th} offset in the sequence of offsets as the xor of the prior offset and the j^{th} basis offset, where j is the number of zero-bits following the last one-bit when the number i is written in binary;

using the block cipher, the key, the nonce, and the length of the message to generate an n-bit final offset;

for each number i between 1 and m-1, xoring the i^{th} message block with the i^{th} offset to determine an i^{th} input block;

for each number i between 1 and m-1, applying the block cipher, keyed by the key, to the i^{th} input block, to determine an i^{th} output block;

for each number i between 1 and m-1, xoring the i^{th} output block with the i^{th} offset to determine an i^{th} ciphertext block;

concatenating the m-1 ciphertext blocks to determine a ciphertext body;

computing an encoded length by encoding the length of the final fragment as an n-bit string;

xoring the encoded length with the final offset to determine a precursor pad;

computing a pad by applying the block cipher, keyed by the key, to the precursor pad;

xoring the final fragment with a portion of the pad to determine a ciphertext fragment having the same length as the final fragment;

computing a padded ciphertext fragment by appending to the ciphertext fragment a sufficient number of zero bits so that the padded ciphertext fragment has n bits;

computing a checksum by xoring together the m-1 message blocks, the pad, and the padded ciphertext fragment;

computing a precursor full tag by xoring together the checksum and the m^{th} offset;

determining a full tag by applying the block cipher, keyed by the key, to the precursor full tag;

computing a tag as a portion of the full tag; and

defining the ciphertext to be the ciphertext body, the ciphertext fragment, and the tag.

69. (Currently amended) ~~An~~ A computer-implemented authenticated-encryption method that uses an n-bit block cipher, a key, and an n-bit nonce to encrypt a message of arbitrary length into a ciphertext of the same length, the method comprising:

partitioning the message into m-1 message blocks and one final fragment, each message block having n bits and the final fragment having between 0 and n bits;

generating m+1 offsets using a sequence shift and xor operations, this sequence of shift and xor operations being applied to a starting value determined using the block cipher, the key, and the nonce;

for each number i between 1 and m-1, xoring the i^{th} message block with the i^{th} offset to determine an i^{th} input block;

for each number i between 1 and m-1, applying the block cipher, keyed by the key, to the i^{th} input block, to determine an i^{th} output block;

for each number i between 1 and m-1, xoring the i^{th} output block with the i^{th} offset to determine an i^{th} ciphertext block;

concatenating the m-1 ciphertext blocks to determine a ciphertext body;

computing an encoded length by encoding the length of the final fragment as an n-bit string;

xoring the encoded length with the m^{th} offset to determine a precursor pad;

computing a pad by applying the block cipher, keyed by the key, to the precursor pad;

xoring the final fragment with a portion of the pad to determine a ciphertext fragment having the same length as the final fragment;

computing a padded ciphertext fragment by appending to the ciphertext fragment a sufficient number of zero bits so that the padded ciphertext fragment has n bits;

computing a checksum by xoring together the m-1 message blocks, the pad, and the padded ciphertext fragment;

computing a precursor full tag by xoring together the checksum and the $(m+1)^{st}$ offset;

determining a full tag by applying the block cipher, keyed by the key, to the precursor full tag;

computing a tag as a portion of the full tag; and

defining the ciphertext to be the ciphertext body, the ciphertext fragment, and the tag.

70 (Currently amended) A computer-readable storage medium storing instructions that when executed by a computer cause the computer to perform an authenticated-encryption method that uses an n-bit block cipher, a key, and an n-bit nonce to encrypt a message of an arbitrary length into a ciphertext of the same length, the method comprising:

partitioning the message into m-1 message blocks and one final fragment, each message block having n bits and the final fragment having between 0 and n bits;

generating m+1 offsets using a sequence shift and xor operations, this sequence of shift and xor operations being applied to a starting value determined using the block cipher, the key, and the nonce;

for each number i between 1 and m-1, xoring the i^{th} message block with the i^{th} offset to determine an i^{th} input block;

for each number i between 1 and m-1, applying the block cipher, keyed by the key, to the i^{th} input block, to determine an i^{th} output block;

for each number i between 1 and m-1, xoring the i^{th} output block with the i^{th} offset to determine an i^{th} ciphertext block;

concatenating the m-1 ciphertext blocks to determine a ciphertext body;

computing an encoded length by encoding the length of the final fragment as an n-bit string;

xoring the encoded length with the m^{th} offset to determine a precursor pad;

computing a pad by applying the block cipher, keyed by the key, to the precursor pad;

xoring the final fragment with a portion of the pad to determine a ciphertext fragment having the same length as the final fragment;

computing a padded ciphertext fragment by appending to the ciphertext fragment a sufficient number of zero bits so that the padded ciphertext fragment has n bits;

computing a checksum by xoring together the m-1 message blocks, the pad, and the padded ciphertext fragment;

computing a precursor full tag by xoring together the checksum and the $(m+1)^{st}$ offset;

determining a full tag by applying the block cipher, keyed by the key, to the precursor full tag;

computing a tag as a portion of the full tag; and

defining the ciphertext to be the ciphertext body, the ciphertext fragment, and the tag.